AND-EXOR 論理式の暗号プロトコルへの応用(計算理論とアルゴリズムの新展開)

AUTHOR(S):

小田切, 太朗; 水木, 敬明; 曽根, 秀昭

数理解析研究所講究録
1489巻 2006年 29-35

AND-EXOR論理式の暗号プロトコルへの応用

東北大学大学院情報科学研究科　小田切 太朗（Taro Otagiri）
Graduate School of Information Sciences, Tohoku University
東北大学情報シナジーセンター　水木 敬明（Takaaki Mizuki）
曽根 秀昭（Hideaki Sone）
Information Synergy Center, Tohoku University

1 はじめに

Feige, Kilian, Naor [3]は，次のような極小モデルにおける安全な計算という問題を考えた．受動的（honest-but-curious）な2人のプレイヤーAliceとBobが存在し，それぞれnビットの秘密情報$a \in \{0,1\}^n$と$b \in \{0,1\}^n$を持っている．AliceとBobは，それぞれの持つ情報を他のプレイヤーに知られることなく，3人目のプレイヤーCarolのみにある論理関数$f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$の計算結果$f(a,b)$を知らせたい．あらかじめAliceとBobはランダムな文字列を共有していてよく，また，AliceとBobは，それぞれ安全な通信路を用いてCarolに対して1回だけメッセージを送信することができるものとする．Carolは，AliceとBobから得られたメッセージにより，求める関数の計算結果のみを知る．また，Alice，BobおよびCarolの計算能力に制限はないものとする．本論文では，このような極小モデルにおける安全な計算について，AND-EXOR論理式を応用した新しいプロトコルを提案する．

1.1 安全な計算の例

極小モデルにおける安全な計算の簡単な例を挙げる．Aliceが1ビットの情報$a \in \{0,1\}$，Bobが1ビットの情報$b \in \{0,1\}$を持っており，Carolが関数としてAliceとBobのビットの排他的論理和$f(a,b) = a \oplus b$を計算したい場合を考える．また，AliceとBobは，あらかじめランダムな1ビット$r \in \{0,1\}$を共有しているものとする．このとき，関数$f(a,b) = a \oplus b$は，以下のプロトコルにより安全に計算することができる．

- Aliceは Carolに対して1ビットのメッセージ$a \oplus r$を送信する．
- Bobは Carolに対して1ビットのメッセージ$b \oplus r$を送信する．
- Carolは$(a \oplus r) \oplus (b \oplus r)$を計算して，関数の計算結果$a \oplus b$を得る．

Aliceから Carolにメッセージを送信する通信路と，Bobから Carolにメッセージを通信する通信路は，ともに安全であると仮定しているので，AliceもBobも相手の持っている秘密情報を知ることはできない．また，AliceおよびBobから Carolに送信されるメッセージは，AliceまたはBobの持つ情報aまたはbとランダムなビットrとの排他的論理和であるので，Carolは関数の計算結果$a \oplus b$のみを得ることになる．

ここで，このプロトコルの性能を評価してみよう．一般に，Aliceが Carolにc_Aビット，Bobが Carolにc_Bビットのメッセージを送信し，AliceとBobの間でc_rビットのランダムな文字列が共有されているとき，そのプロトコルを$(c_A, c_B; c_r)$-プロトコルと呼ぶことにする．したがって，上

述のEXOR関数 $f(a,b)=a\oplus b$ を安全に計算するプロトコルでは，AliceがCarolに1ビット，BobがCarolに1ビットを送信し，AliceとBobの間で1ビットのランダムな文字列が共有されているので，このプロトコルは $(1,1;1)$-プロトコルである．

1.2 既知のプロトコル

極小モデルにおける安全な計算のプロトコルとして，Feige, Kilian, Naor [3]は，全ての関数を計算可能なものを与えている．そのプロトコルでは，関数 $f:\{0,1\}^n\times\{0,1\}^n\to\{0,1\}$ を計算するとき，AliceからCarolに対して 2^n ビット，BobからCarolに対して $n+1$ ビットのメッセージが送信され，AliceとBobの間では 2^n+n ビットのランダムな文字列を共有している必要がある．すなわち，彼らが与えたプロトコルは，$(2^n,n+1;2^n+n)$-プロトコルである．

また，Feigeら[3]は，関数のあるクラスに対して効率の良いプロトコルを提案した．具体的には，$f\in\mathbf{NL}$ となる関数に対して，彼らのプロトコルは，n の多項式である c_A，c_B，c_r に対して $(c_A,c_B;c_r)$-プロトコルである．

IshaiとKushilevitz [9]は，PSMプロトコルと呼ばれるプロトコルを提案した．このプロトコルは，効率的に計算できる関数のクラスを，NLから $\mathbf{Mod}_k\mathbf{L}$, $\mathbf{C}_=\mathbf{L}$, #L および DiffL まで拡張した．

1.3 提案プロトコル

本論文では，極小モデルにおける安全な計算を実現する新しいプロトコルを提案する．提案プロトコルは全ての論理関数に対して計算可能であり，単純なプロトコルである．提案プロトコルの詳細については2節で述べるが，ここでは簡単に提案プロトコルの概要を説明する．

本論文で提案するプロトコルは，安全な計算にExclusive-or-Sum-of Products（ESOP）表現の技術を応用している．ESOPとは，任意の論理関数を複数の積項の排他的論理和の結合として表したものである．提案プロトコルでは，計算したい関数 f をESOPで表現し，ESOP表現の各積項ごとに計算を行い，最後に全体の排他的論理和を計算することによって求める関数の計算結果を得る．提案プロトコルは，すべての論理関数を計算することができ，関数 $f:\{0,1\}^n\times\{0,1\}^n\to\{0,1\}$ を計算する際の性能は，その関数 f をESOPで表現した式の積項数に依存する．関数 f をESOPで表現したものを F，F に含まれる積項数を $\tau(F)$ とすると，提案プロトコルは $(2\tau(F),\tau(F)+1;3\tau(F))$-プロトコルである．

上述の通り，提案プロトコルにおいて必要なメッセージ等のビット数は，関数 f のESOP表現 F に含まれる積項数に依存している．したがって，積項数が小さくなるほど，提案プロトコルは効率的になる．関数をESOPで表現する場合，1つの関数に対して，無数のESOP表現が存在する．ESOPに関する研究は論理回路設計の分野で長年行われてきており，関数をESOPで表現した際に含まれる積項数をできる限り小さくしようとするヒューリスティックなESOP簡単化の研究が，盛んに行われている[12]．しかし，ESOPの技術を暗号プロトコルの設計に応用した例はなく，本提案プロトコルは画期的なプロトコルである．

1.4 関連研究

Yao [17] のプロトコル以来現在まで，安全な計算に関して，非常に多くの研究が行われている [7]．本論文で考えているサードパーティーモデルに関連する研究例として，Cachin と Camenisch の研究が挙げられる [1]．彼らの 2 プレイヤーによる安全な計算のプロトコルでは，Carol は計算に加わることができるが，通常のプロトコル実行の場合には Carol は参加せず，Alice と Bob 自身が関数 $f(a,b)$ を安全に計算する．また，各プレイヤー間の通信が一方向のモデルに関連しては，文献 [2, 5, 6, 10] が挙げられる．

2 新しいプロトコルの提案

本節では，極小モデルにおける安全な計算を実現する新しいプロトコルを提案する．

2.1 プロトコル本体

Alice と Bob はそれぞれ秘密情報 $a \in \{0,1\}^n$ と $b \in \{0,1\}^n$ を持っている．ある論理関数 $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ について，Alice と Bob の持つ情報 a,b を他のプレイヤーに知られることなく，関数の計算結果 $f(a,b)$ のみを Carol だけが知るようにしたい．提案プロトコルでは，以下の手順に従って関数 f を安全に計算する．

1. 計算したい関数 f を次のような Exclusive-Or-Sum-of Products (ESOP) で表現する．

$$f(a,b) = A_1(a)B_1(b) \oplus A_2(a)B_2(b) \oplus \cdots \oplus A_t(a)B_t(b)$$

   ここで，$A_i, B_i\ (1 \le i \le t)$ は，$A_i : \{0,1\}^n \to \{0,1\}$，$B_i : \{0,1\}^n \to \{0,1\}$ を満足するとする．(2.2.1 節でこのステップ 1 の ESOP 表現の詳細を述べる．)

2. ESOP 表現の各積項 $A_i(a)B_i(b)$ について，あるサブプロトコルを実行する．このとき，Carol は計算結果として $A_i(a)B_i(b) \oplus k_i$ を得る．ここで，$k_i \in \{0,1\}$ は乱数であり，Bob のみがその値を知っている．すなわち，各々の積項について，Carol は，Bob のみが知る鍵 k_i によって暗号化された積項の計算結果を得ることになる．(このステップ 2 のサブプロトコルについては，2.2.2 節で詳細を述べる．)

3. ステップ 2 の終了後，Carol は 1 ビットの値 $A_i(a)B_i(b) \oplus k_i$ を t 個持っている．また，Bob は各積項の計算結果の暗号化に用いられている鍵 $k_1, k_2, \cdots, k_t$ をすべて知っている．Bob は Carol に 1 ビットのメッセージ $k_1 \oplus k_2 \oplus \cdots \oplus k_t$ を送信する．Carol は，Bob から受け取ったメッセージと，ステップ 2 で Alice から受け取った値を用いて

$$\bigoplus_{i=1}^{t} \left(A_i(a)B_i(b) \oplus k_i \right) \oplus (k_1 \oplus k_2 \oplus \cdots \oplus k_t)$$

   を計算し，求める計算結果

$$f(a,b) = A_1(a)B_1(b) \oplus A_2(a)B_2(b) \oplus \cdots \oplus A_t(a)B_t(b)$$

   を得る．

ステップ 1 における f の ESOP 展開の積項数が t であるとき，提案プロトコルは $(2t, t+1; 3t)$-プロトコルである．したがって，提案プロトコルにおいて必要なメッセージ等のビット数は，計算したい関数 f を ESOP で表現した式に含まれる積項数に依存している．

なお，紙面の都合上，本プロトコルの安全性の証明は割愛する．

2.2 プロトコルの構成要素

本節では，提案プロトコルの構成要素を与える．

2.2.1 Exclusive-Or-Sum-of Products (ESOP) の応用

本提案プロトコルでは，ステップ 2 に示した通り，安全な計算に ESOP を応用している．

ESOP は論理回路設計の分野で研究が進んでおり，任意の論理関数を ESOP で表した際の ESOP を構成する積項数を小さくする研究，ESOP 簡単化・最小化の研究が長年行なわれてきている．任意の ESOP の積項数を最小にする効率的なアルゴリズムはまだ見付かっていないが，効率の良いヒューリスティックな ESOP 簡単化アルゴリズムが多数提案されている [4, 11, 13, 14, 18]．また，論理式に用いられている変数が少ない場合のみに関しては，ESOP の積項数を最小化するアルゴリズムも提案されている [8, 15, 16]．ESOP 簡単化に関しては現在も研究が進められている分野である．

これまでの ESOP に関する研究では，変数が 2 値で多変数の場合について多く研究されてきた．2 値多変数の ESOP 表現の最も有名な例として，正極性 Reed-Muller 展開が挙げられる．2 値多変数の ESOP 展開に関しては，多くのヒューリスティックな ESOP 最小化・簡単化アルゴリズムが提案されている [4, 8, 15, 18]．また，変数が多値の場合についても研究されており，いくつかの簡単化アルゴリズムが提案されている．特に，4 値の場合について，2 ビット入力デコーダ付 PLA の設計に関連して，多く研究が行なわれてきた [11, 13]．また，1 つの変数のみ多値の場合についても分析されている [16]．

ここから，提案プロトコルへの ESOP の応用について考える．考えているのは極小モデルにおける安全な計算であり，Alice，Bob，Carol の 3 プレイヤーにより関数

$$f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$$

を計算したい．関数の入力となる変数は Alice が持つ情報 $a \in \{0,1\}^n$，Bob が持つ情報 $b \in \{0,1\}^n$ である．Alice，Bob はそれぞれ 2 値 n ビットの情報を関数の入力情報として持っている．よって，2^n 値 2 変数の関数の ESOP を考えればよい．このとき，簡単化（あるいは最小化）アルゴリズムを用いて関数 f を ESOP で表現すると，以下のような論理式を得る．

$$f(a,b) = A_1(a)B_1(b) \oplus A_2(a)B_2(b) \oplus \cdots \oplus A_t(a)B_t(b)$$

ただし，$A_i, B_i (1 \le i \le t)$ は，$A_i : \{0,1\}^n \to \{0,1\}$，$B_i : \{0,1\}^n \to \{0,1\}$ を満足する．

2.2.2 サブプロトコル

ここでは，本プロトコルのステップ 2 における，サブプロトコルを与える．

Alice と Bob はそれぞれ秘密情報 $a \in \{0,1\}^n$ と $b \in \{0,1\}^n$ を持っているとする．関数 A, B を，$A : \{0,1\}^n \to \{0,1\}$，$B : \{0,1\}^n \to \{0,1\}$ とする．また，Alice と Bob は Carol に対して，1 ビット $A(a)B(b) \oplus k$ を知らせたいとする．ここで，k はランダムな鍵で，Bob だけがその値を知っている．本節では，これを実現するサブプロトコルを与える．

計算手順を示す前に，shift と get という操作を定義する．2 ビット (x,y) が与えられたとき，shift および get を以下のように定義する．

$$\begin{aligned} \mathrm{shift}^0(x,y) &= (x,y); \\ \mathrm{shift}^1(x,y) &= (y,x); \\ \mathrm{get}^0(x,y) &= x; \\ \mathrm{get}^1(x,y) &= y. \end{aligned}$$

$\mathrm{shift}^0(x,y)$ はそのまま値を返す操作，$\mathrm{shift}^1(x,y)$ は (x,y) の値を入れ替えて値を返す操作である．また，$\mathrm{get}^0(x,y)$ は 2 ビットの情報のうち，1 番目の情報を取り出す操作，$\mathrm{get}^1(x,y)$ は 2 ビットの情報のうち，2 番目の情報を取り出す操作である．

以下に，Carol に情報 $A(a)B(b) \oplus k$ を知らせる手順を説明する．ここで，Alice と Bob の間で 3 ビットのランダムな文字列 $((k^0,k^1),s)$ を共有しているとする．k^0 および k^1 は，Carol に送信するメッセージを暗号化するために用いられる鍵である．また，s は，Carol に送信するメッセージの値をシフトするために用いられる値である．

- Alice は Bob の持つ情報 $b \in \{0,1\}^n$ の値を知らないので，$B(b) = 0$ と $B(b) = 1$ の両方の可能性を考え，以下のような 2 ビットのメッセージを用意する．

  $$(A(a) \cdot 0, A(a) \cdot 1) = (0, A(a))$$

  Alice は Bob と共有している鍵 (k^0,k^1) を用いてメッセージを暗号化する．

  $$(k^0, A(a) \oplus k^1)$$

  さらに，Alice は Bob と共有している情報 s を用いてこれら 2 ビットのメッセージの値をシフトして得られる

  $$\mathrm{shift}^s(k^0, A(a) \oplus k^1)$$

  を Carol に送信する．すなわち，Alice は Carol に対して以下のようなメッセージを送信する．

  $$\begin{cases} (k^0, A(a) \oplus k^1) & \text{if } s = 0; \\ (A(a) \oplus k^1, k^0) & \text{if } s = 1 \end{cases}$$

- Bob は Alice が Carol に送信した 2 ビットのメッセージ $\mathrm{shift}^s(k^0, A(a) \oplus k^1)$ について，2 ビットのうちどちらが正しい値であるかを知っている．すなわち，$B(b) = s = 0$ または $B(b) = s = 1$ のとき，メッセージ $\mathrm{shift}^s(k^0, A(a) \oplus k^1)$ のうち 1 番目の値が正しい値 $(A(a)B(b) \oplus k^{B(b)})$ であり，それ以外のときは 2 番目の値が正しい値である．よって Bob は，Carol に対して 1 ビットのメッセージ

  $$B(b) \oplus s$$

を送信する．Carolはこの$B(b)\oplus s$の値によって，Aliceから受け取った2ビットのメッセージのうち，どちらが正しい値であるかを知ることができる．

- CarolはAliceとBobから受け取ったメッセージを用いて，

$$\mathrm{get}^{B(b)\oplus s}(\mathrm{shift}^{s}(k^{0},A(a)\oplus k^{1}))$$

の操作を行なう．よって，Carolは情報$A(a)B(b)\oplus k^{B(b)}$を得ることができる．

ランダムな鍵$k^{B(b)}$の値はBobのみが知っているため，このサブプロトコルは目的を達する．なお，このサブプロトコルは，Alice-Carol間で2ビット，Bob-Carol間で1ビット，Alice-Bob間で3ビット必要であるので，(2,1;3)-プロトコルである．

3 むすび

本論文では，極小モデルにおける安全な計算という問題に対して，そのような計算を実現する新しいプロトコルを提案した．提案プロトコルは，安全な計算にESOP表現を応用したものであり，全ての関数に対して計算可能である．また，既存プロトコルよりも単純なプロトコルであり，暗号理論の分野にESOPを導入した画期的なものである．関数のESOP表現式に含まれる積項数をtとすると，提案プロトコルは$(2t,t+3;3t)$-プロトコルである．Feige, Kilian, Naor [3]による既存プロトコルは，$(2^n,n+1;2^n+n)$-プロトコルであり，関数計算に必要なビット数は入力情報のビット数に依存しているが，提案プロトコルに必要なビット数は，関数およびESOP簡単化・最小化アルゴリズムの性能に依存している．そのため，提案プロトコルの性能は計算する関数やESOP簡単化アルゴリズムによって向上する．また，今後のESOP研究によりさらにプロトコル性能が向上する可能性があり，論理回路設計分野の新たなモチベーションにもなり得ると考えられる．

参考文献

[1] C. Cachin and J. Camenisch, "Optimistic fair secure computation," Proc. CRYPTO 2000, Lecture Notes in Computer Science, vol. 1880, pp. 93–111, Springer-Verlag, 2000.

[2] C. Cachin, J. Camenisch, J. Kilian, and J. Müller, "One-round secure computation and secure autonomous mobile agents," Proc. ICALP 2000, Lecture Notes in Computer Science, vol. 1853, pp. 512–523, Springer-Verlag, 2000.

[3] U. Feige, J. Kilian, and M. Naor, "A minimal model for secure computation," Proceedings of the 26th ACM Symposium on Theory of Computing (STOC '94), pp. 554–563, 1994.

[4] H. Fleisher, M. Tavel, and J. Yeager, "A computer algorithm for minimizing Reed-Muller canonical forms," IEEE Transactions on Computers, vol. 36, no. 2, pp. 247–250, 1987.

[5] A. Gál and A. Rosén, "A theorem on sensitivity and applications in private computation," SIAM Journal on Computing, vol. 31, no. 5, pp. 1424–1437, 2002.

[6] A. Gál and A. Rosén, "Lower bounds on the amount of randomness in private computation," Proceedings of the 35th ACM Symposium on Theory of Computing (STOC '03), pp. 659–666, 2003.

[7] O. Goldreich, "Foundations of Cryptography II: Basic Applications," Cambridge University Press, Cambridge, 2004.

[8] T. Hirayama, Y. Nishitani, and T. Sato, "A faster algorithm of minimizing AND-EXOR expressions," IEICE Trans. Fundamentals, vol. E85-A, no. 12, pp. 2708-2714, 2002.

[9] Y. Ishai and E. Kushilevitz, "Private simultaneous messages protocols with applications," Proceedings of the fifth Israel Symposium on the Theory of Computing Systems (ISTCS '97), pp. 174–183, 1997.

[10] E. Kushilevitz, R. Ostrovsky, and A. Rosén, "Characterizing linear size circuits in terms of privacy," Journal of Computer and System Sciences, vol. 58, no. 1, pp. 129–136, 1999.

[11] T. Sasao, "EXMIN2: A Simplification Algorithm for Exclusive-OR-Sum-of Products Expressions for Multiple-Valued-Input Two-Valued-Output Functions," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 12, no. 5, pp. 621–632, 1993.

[12] T. Sasao, "Switching Theory for Logic Synthesis," Kluwer Academic Publishers, Boston, MA, 1999.

[13] T. Sasao and P. Besslich, "On the complexity of mod-2 sum PLA's," IEEE Transactions on Computers, vol. 39, no. 2, pp. 262–266, 1990.

[14] N. Song and M. A. Perkowski, "Minimization of exclusive sum-of-products expressions for multiple-valued input, incompletely specified functions," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 15, no. 4, pp. 385–395, 1996.

[15] S. Stergiou and G. Papakonstantinou, "Exact minimization of ESOP expressions with less than eight product terms," Journal of Circuits, Systems and Computers, vol. 13, no. 1, pp. 1–15, 2004.

[16] S. Stergiou, D. Voudouris, and G. Papakonstantinou, "Multiple-value exclusive-or sum-of-products minimization algorithms," IEICE Trans. Fundamentals, vol. E87-A, no. 5, pp. 1226–1234, 2004.

[17] A. Yao, "Protocols for secure computations," Proceedings of the 23th IEEE Symposium on Foundations of Computer Science (FOCS '82), pp. 160–164, 1982.

[18] Y. Ye and K. Roy, "An XOR-based decomposition diagram and its application in synthesis of AND/XOR networks," IEICE Trans. Fundamentals, vol. E80-A, no. 10, pp. 1742–1748, 1997.